NexusScout: An Advanced Location-Based Application on a Distributed, Open Mediation Platform

Daniela Nicklas *Matthias Grossmann* *Thomas Schwarz*

University of Stuttgart, Institute of Parallel and Distributed Systems
Universitätsstraße 38, 70569 Stuttgart, Germany
firstname.lastname@informatik.uni-stuttgart.de

This demo shows several advanced use cases of location-based services and demonstrates how these use cases are facilitated by a mediation middleware for spatial information, the Nexus Platform. The scenario shows how a mobile user can access location-based information via so called Virtual Information Towers, register spatial events, send and receive geographical messages or find her friends by displaying other mobile users. The platform facilitates these functions by transparently combining spatial data from a dynamically changing set of data providers, tracking mobile objects and observing registered spatial events.

1. Introduction

Location-based services (LBS) are a new and exciting application domain that leads to new challenges in data management. Advanced location-based applications adapt to the user's context. Thus they need to determine this context which in our case can be retrieved from a context model of the user's environment, imitating the real world and connecting it to virtual information. The context model involves spatial data as well as traditional information from databases or webpages, augmented with a spatial reference.
The research project Nexus at the University of Stuttgart focuses on world models for mobile, context-aware applications. During the last three years, a team of researchers from four departments (distributed systems, database and applications, photogrammetry and communication networks) has built an open platform to manage a distributed world model and to make it accessible to location-based services. In our demonstrator application, the NexusScout, we have integrated advanced use cases of LBS and have evaluated how a middleware platform can support them.

Proceedings of the 29th VLDB Conference,
Berlin, Germany, 2003

2. The Application: NexusScout

NexusScout is a tool for a mobile user to orientate oneself and to access geo-referenced information. With our sample data, we can show the following scenario: a group of students visits the university campus in Stuttgart. They can use the NexusScout to get information about points of interest like university buildings, and use more advanced functions as described below. Some of the members of the group have different interests and so they split up. This allows us to demonstrate some use cases of the NexusScout that require more than one user, e.g. "display the position of the other members of my group". We will show two NexusScouts on two different notebooks that both use the Nexus platform (installed on a third one). In the following, we describe the use cases that we will show in the demo.

2.1 Display Map. For the orientation of the mobile user, a map of the surrounding area is helpful. The map displays the current position of the user and can be switched between different zoom levels. For the NexusScout, the map is of great importance: apart from showing buildings and streets it visualizes all the relevant location-based information that is gathered from different servers and shows always the most recent situation.

2.2 Virtual Information Towers. Figure 1 shows little pillars on the map. They are called Virtual Information Towers (VIT) and serve as metaphors for web pages (posters) relevant at the location of the VITs. A VIT has an area of visibility and appears on the map when the user enters this area [LKRF99].

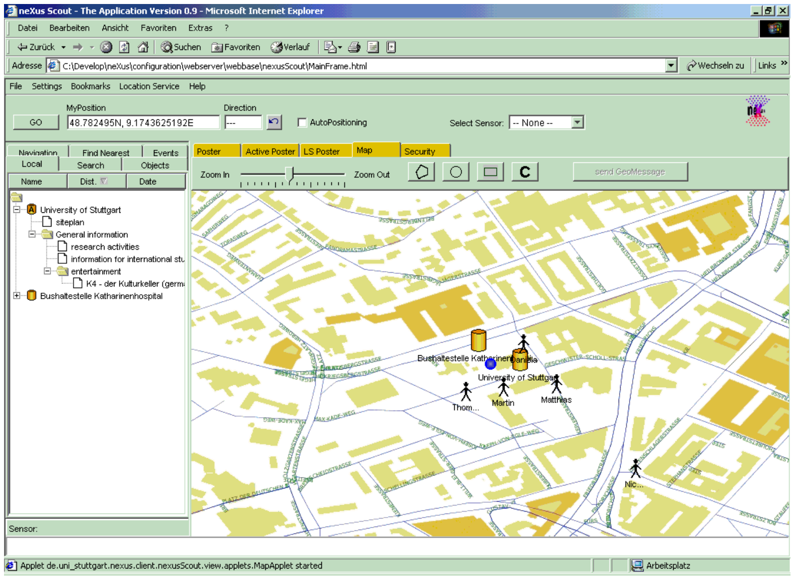

Figure 1: Screenshot NexusScout - map with Virtual Information Towers and mobile users

On the left side of the screen, the poster hierarchies of all visible VITs are shown. When clicking on a poster, the NexusScout switches to the poster view and displays the chosen web page. In the demo, we will show how the changing set of visible VITs is displayed on the map and how the posters can be accessed.

2.3 Mobile Objects. The NexusScout can display the position of other mobile users, e.g. the other members of the visiting group (see Figure 1). They are updated every 5 seconds. In the demo, we will display the position of the other real NexusScout user as well as several simulated users.

2.4 Spatial Events. With the NexusScout a user can receive notifications of previously defined spatial events [BaRo02]. When they occur, she gets a notification. In the demo, we will show two different types of spatial events:

- OnEnterArea: the user selects an area on the map with the mouse. When she walks to the selected area, she will get a notification. We will define such an event for the library.
- OnMeeting: the user chooses another mobile user (see Use Case Mobile Objects). She gets a notification when she comes close to this user.

2.5 Positioning. A precondition for location-based services is the ability of the application to determine its position. The NexusScout provides three positioning methods, which can be selected manually:

1. GPS: The NexusScout gets the position from an external GPS device. The accuracy is 5-10m, and it can only be used outdoors.
2. Infrared: Infrared beacons send unique IDs. The NexusScout maps this ID to a geographical position using a table. The accuracy is 3-10m (depending on the visibility of the infrared beacon). The technique can be used wherever such beacons can be placed.
3. Manually: the user determines her position by clicking on the map. The accuracy is 1-3m (depending on the zoom level) and the technique can be used ubiquitously.

In the demo we will show techniques 2 and 3. Because our sample data is from our home town Stuttgart, we will map the IDs to geographical positions in our test area and this way simulate the users moving around on the campus.

2.6 Geo Messaging. GeoCast is a distributed service that is used to send messages to a geographical area. All users that are currently in this area receive the message [CRD02]. We have integrated this function into the NexusScout: similar to OnEnterArea events (see Use Case Spatial Events), the first user specifies an area and types in a message. If the position of the second NexusScout user is in that area, she will receive the message.

3. The Nexus Platform

The Nexus Platform is a distributed infrastructure that supports applications like the NexusScout [HKL+99]. We call it an *open platform* because the data providers (*Spatial Model Servers*) are not known to the application, but are dynamically chosen using a registry. Like in the world wide web, everybody can add a new server and only has to register it to make it accessible for applications. The federation tier of the platform lets applications perceive the distributed data as a single context model, which we call the *Augmented World Model (AWM)* [NiMi01]. The model is object-based, has an hierarchical inheritance relationship, and the majority of

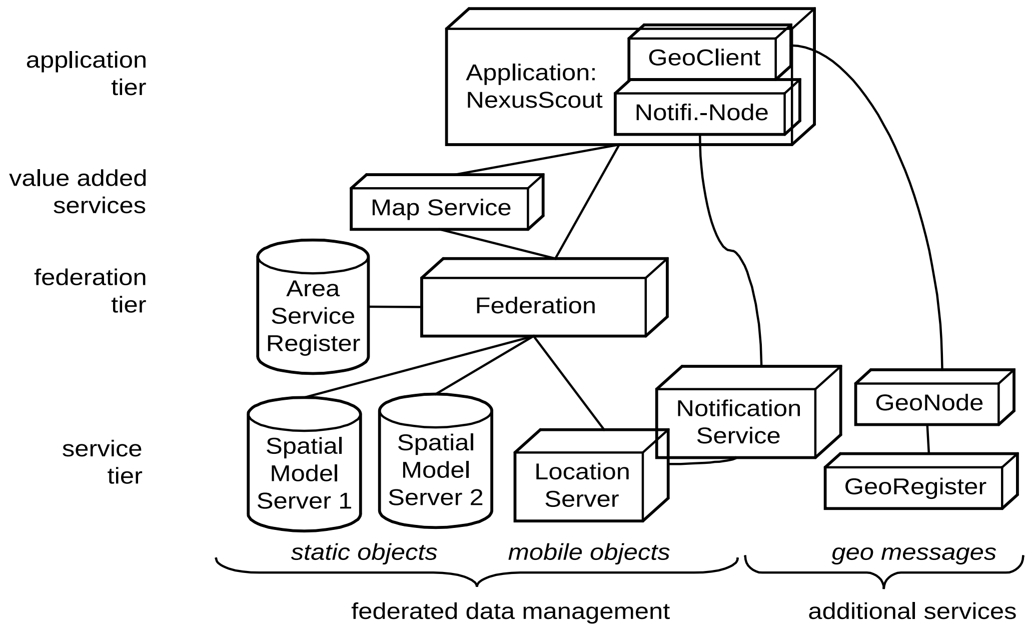

Figure 2: Architecture of Demonstrated Nexus Platform

its objects possess at least one spatial attribute (e.g., position or extent).

In Figure 2, we highlight the architecture of the Nexus Platform as it is used in the demonstration. Most components exchange data in several XML formats and have SOAP interfaces. In order to minimize the number of computers for the demonstration, we will install several components on one notebook, but in principle each component would run on its own computer. All components are implemented in pure Java. We will now briefly describe the components of the Nexus Platform and illustrate how they contribute to the demonstration.

3.1 Spatial Model Server. Data providers use Spatial Model Servers to store spatial, non-moving objects like buildings and streets, but also Virtual Information Towers with their poster hierarchy (see Use Case Virtual Information Towers), their geographical position and their visibility area. These objects can be retrieved via a simple query language called Augmented World Query Language (AWQL) [NGS+01]. The main part of an AWQL query is a restriction: a boolean expression which must be true for an object to be included in the result set. Compared to SQL, AWQL does not support joins, but allows more heterogeneous result sets. Figure 3 shows a query that retrieves all building-objects (and objects of sub-classes of building) within the specified area. The result is serialized in an XML format called AWML.

To actually store data, an arbitrary system may be used, as long as a wrapper is provided, which maps its interface to AWQL. We are using IBMs DB2 and SpatialExtender. All non-moving data in the demonstration (except for web pages) is retrieved from the two installed Spatial Model Servers.

```
<?xml version="1.0" encoding="UTF-8"?>
<awql xmlns:xsi="http://www.w3.org/2001/XMLSchema
-instance" xsi:noNamespaceSchemaLocation="http://
pcquaddmg.informatik.uni-stuttgart.de:8080/spase/
awql.xsd">
 <restriction>
  <and>
   <equal>
    <attr name="type"/>
    <nexusdata><Table>Building</Table></nexusdata>
   </equal>
   <overlaps>
    <attr name="extent"/>
    <nexusdata><WKT> POLYGON ((9.106228 48.745541,
9.107140 48.745497, 9.107068 48.744789,
9.106152 48.744833, 9.106228 48.745541))
    </WKT></nexusdata>
   </overlaps>
  </and>
 </restriction>
</awql>
```

Figure 3: AWQL query

3.2 Location Server and Notification Service. The Location Server manages the positions of mobile objects in an efficient main memory structure [LeKu99]. The position data is characterized by a high update rate and a low persistency as mobile applications frequently send position updates. For such data a Spatial Model Server, which is based on a full-grown database system, is too inflexible. Additionally, the Location Server observes spatial events. If an event occurs, the event component informs the Notification Service which sends a message to all applications interested in this event. The Location Service is mandatory for the Use Case Spatial Events and the Use Case Mobile Objects.

3.3 Federation Tier. The federation tier mediates between the servers of data providers including the Location Server and the application. It provides a homogenized view on the heterogeneous data sets of different servers. It can easily be duplicated for scalability as it does not store any data.

Spatial Model Servers have to register at the Area Service Register (ASR), which is comparable to a spatially enhanced DNS, allowing the Federation to process queries as follows. It extracts the query region and the list of requested object types from the query, and, based on this information, retrieves a list of relevant servers from the ASR. Then the original query is forwarded to these servers (both the Federation and the servers use AWQL as query language). Results from all servers are collected and merged into a single result. Multiple representations of the same real-world object are merged by creating a new object that contains all attributes of the corresponding representations.

In the demo, we will show this functionality by registering and unregistering one of the two Spatial Model Servers over a simple GUI of the ASR. Unregistering a Spatial Model Server will cause objects hosted by this server to disappear from the Nexus-Scout's map.

3.4 Value Added Services. The Nexus Platform hosts value added services that complement the basic query functionality of the Federation. By our definition, value added services use the federated data and do some further processing, thus delivering enhanced data to the application.

In the demo, we will show one value added service, the Map Service. The application specifies the area the map shall cover and the types of objects it shall contain. The Map Service queries the Federation and renders the results in a map, which is returned as an image to the application, see Figure 1.

3.5 Additional Services. Beside the federated data management, location-based applications can of course use additional services like the WWW or the above described Geo Messaging.

4. Outlook and Challenges

Several challenges arise from the openness of the platform and from the application domain. The platform needs to provide an integrated view on the data stored at different providers with varying availability. Two providers can survey the same real world object differently. Either both representations augment each other or they conflict, in any case the application should receive a merged representation. Currently XML technology allows us to merge representations on a syntactical level. We are working on lifting this to a semantic level by integrating application domain specific knowledge into the merge process.

5. Acknowledgements

We want to thank our project partners from the departments of Distributed Systems, Photogrammetry and Communication Networks, which have largely contributed to the implementation of the platform and the NexusScout. This work was made possible with the help of several study projects and theses. The Nexus project was funded 1999-2002 by the German Research Association (DFG) the under grant 200989 and is now continued as Collaborative Research Center (SFB) 627.

References

[BaRo02] Bauer, Martin; Rothermel, Kurt: **Towards the Observation of Spatial Events in Distributed Location-Aware Systems.** In: Wagner, Roland (ed.): Proceedings of the 22nd International Conference on Distributed Computing Systems Workshops, 2002

[CRD02] Coschurba, Peter; Rothermel, Kurt; Dürr, Frank: **A fine grained addressing concept for GeoCast.** In: Schmeck, Hartmut; Unger, Theo; Wolf, Lars (ed.): Proceedings of the International Conference on Architecture of Computing Systems (ARCS), 2002

[HKL+99] Hohl, F.; Kubach, U.; Leonhardi, A.; Rothermel, K.; Schwehm, M. **Next Century Challenges: Nexus - An Open Global Infrastructure for Spatial-Aware Applications.** In: Proceedings of the Fifth Annual International Conference on Mobile Computing and Networking (MobiCom), 1999

[LeKu99] Leonhardi, A.; Kubach, U.: **An Architecture for a Distributed Universal Location Service.** In: Proceedings of the European Wireless '99 Conference, 1999

[LKRF99] Leonhardi, A.; Kubach, U.; Rothermel, K.; Fritz, A. **Virtual Information Towers - A Metaphor for Intuitive, Location-Aware Information Access in a Mobile Environment.** In: Proceedings of the Third International Symposium on Wearable Computers (ISCW), 1999

[NGS+01] D. Nicklas, M. Grossmann, T. Schwarz, S. Volz, B. Mitschang: **A Model-Based, Open Architecture for Mobile, Spatially Aware Applications.** In: Jensen, Christian S. (ed.); Schneider, Markus (ed.); Seeger, Bernhard (ed.); Tsotras, Vassilis J. (ed.): Proceedings of the 7th International Symposium on Spatial and Temporal Databases (SSTD), 2001

[NiMi01] D. Nicklas, B. Mitschang: **The Nexus Augmented World Model: An Extensible Approach for Mobile, Spatially-Aware Applications.** In: Wang, Yingxu (ed.); Patel, Shushma (ed.); Johnston, Ronald (ed.): Proceedings of the 7th International Conference on Object-Oriented Information Systems (OOIS), 2001